Learning Associative Markov Networks

Ben Taskar BTASKAR@CS.STANFORD.EDU
Vassil Chatalbashev VASCO@CS.STANFORD.EDU
Daphne Koller KOLLER@CS.STANFORD.EDU
Computer Science Department, Stanford University, Stanford, CA

Abstract

Markov networks are extensively used to model complex sequential, spatial, and relational interactions in fields as diverse as image processing, natural language analysis, and bioinformatics. However, inference and learning in general Markov networks is intractable. In this paper, we focus on learning a large subclass of such models (called *associative Markov networks*) that are tractable or closely approximable. This subclass contains networks of discrete variables with K labels each and clique potentials that favor the same labels for all variables in the clique. Such networks capture the "guilt by association" pattern of reasoning present in many domains, in which connected ("associated") variables tend to have the same label. Our approach exploits a linear programming relaxation for the task of finding the best joint assignment in such networks, which provides an approximate quadratic program (QP) for the problem of learning a margin-maximizing Markov network. We show that for associative Markov network over binary-valued variables, this approximate QP is guaranteed to return an optimal parameterization for Markov networks of arbitrary topology. For the non-binary case, optimality is not guaranteed, but the relaxation produces good solutions in practice. Experimental results with hypertext and newswire classification show significant advantages over standard approaches.

1. Introduction

Numerous classification methods have been developed for the principal machine learning problem of assigning to a single object one of K labels consistent with its properties. Many classification problems, however, involve sets of related objects whose labels must also be consistent with each other. In hypertext or bibliographic classification, labels of linked and co-cited documents tend to be similar (Chakrabarti et al., 1998; Taskar et al., 2002). In proteomic analysis, location and function of proteins that interact are often highly correlated (Vazquez et al., 2003). In image processing, neighboring pixels exhibit local label coherence in denoising, segmentation and stereo correspondence (Besag, 1986; Boykov et al., 1999a).

Markov networks compactly represent complex joint distributions of the label variables by modeling their local interactions. Such models are encoded by a graph, whose nodes represent the different object labels, and whose edges represent direct dependencies between them. For example, a Markov network for the hypertext domain would include a node for each webpage, encoding its label, and an edge between any pair of webpages whose labels are directly correlated (e.g., because one links to the other).

There has been growing interest in training Markov networks for the purpose of collectively classifying sets of related instances. The focus has been on discriminative training, which, given enough data, generally provides significant improvements in classification accuracy over generative training. For example, Markov networks can be trained to maximize the conditional likelihood of the labels given the features of the objects (Lafferty et al., 2001; Taskar et al., 2002). Recently, maximum margin-based training has been shown to additionally boost accuracy over conditional likelihood methods and allow a seamless integration of kernel methods with Markov networks (Taskar et al., 2003a).

The chief computational bottleneck in this task is inference in the underlying network, which is a core subroutine for all methods for training Markov networks. Probabilistic inference is NP-hard in general, and requires exponential time in a broad range of practical Markov network structures, including grid-topology networks (Besag, 1986). One can address the tractability issue by limiting the structure of the underlying network. In some cases, such as the the quad-tree model used for image segmentation (Bouman & Shapiro, 1994), a tractable structure is determined in advance. In other cases (e.g., (Bach & Jordan, 2001)),

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004.

the network structure is learned, subject to the constraint that inference on these networks is tractable. In many cases, however, the topology of the Markov network does not allow tractable inference. In the hypertext domain, the network structure mirrors the hyperlink graph, which is usually highly interconnected, leading to computationally intractable networks.

In this paper, we show that optimal learning is feasible for an important subclass of Markov networks — networks with *attractive potentials*. This subclass, which we call *associative Markov networks (AMNs)*, contains networks of discrete variables with K labels each and arbitrary-size clique potentials with K parameters that favor the same label for all variables in the clique. Such positive interactions capture the "guilt by association" pattern of reasoning present in many domains, in which connected ("associated") variables tend to have the same label. AMNs are a natural fit for object recognition and segmentation, webpage classification, and many other applications.

Our analysis is based on the maximum margin approach to training Markov networks, presented by Taskar *et al.* (2003a). In this formulation, the learning task is to find the Markov network parameterization that achieves the highest confidence in the target labels. In other words, the goal is to maximize the margin between the target labels and any other label assignment. The inference subtask in this formulation of the learning problem is one of finding the best joint (MAP) assignment to all of the variables in a Markov network. By contrast, other learning tasks (e.g., maximizing the conditional likelihood of the target labels given the features) often require that we compute the posterior probabilities of different label assignments, rather than just the MAP.

The MAP problem can naturally be expressed as an integer programming problem. We show how we can approximate the maximum margin Markov network learning task as a quadratic program that uses a linear program (LP) relaxation of this integer program. This quadratic program can be solved in polynomial time using standard techniques. We show that whenever the MAP LP relaxation is guaranteed to return integer solutions, the approximate max-margin QP provides an optimal solution to the max-margin optimization task. In particular, for associative Markov networks over binary variables ($K = 2$), this linear program provides exact answers. For the non-binary case ($K > 2$), the approximate quadratic program is not guaranteed to be optimal, but our empirical results suggest that the solutions work well in practice. To our knowledge, our method is the first to allow training Markov networks of arbitrary topology.

2. Markov Networks

We restrict attention to networks over discrete variables $\mathbf{Y} = \{Y_1, \ldots, Y_N\}$, where each variable corresponds to an object we wish to classify and has K possible labels: $Y_i \in \{1, \ldots, K\}$. An assignment of values to $\mathbf{Y}$ is denoted by $\mathbf{y}$. A Markov network for $\mathbf{Y}$ defines a joint distribution over $\{1, \ldots, K\}^N$.

A Markov network is defined by an undirected graph over the nodes $\mathbf{Y} = \{Y_1, \ldots, Y_N\}$. In general, a Markov network is a set of *cliques* $\mathcal{C}$, where each clique $c \in \mathcal{C}$ is associated with a subset Y_c of $\mathbf{Y}$. The nodes Y_i in a clique c form a fully connected subgraph (a clique) in the Markov network graph. Each clique is accompanied by a *potential* $\phi_c(Y_c)$, which associates a non-negative value with each assignment $\mathbf{y}_c$ to Y_c. The Markov network defines the probability distribution:

$$P_\phi(\mathbf{y}) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{y}_c)$$

where Z is the *partition function* given by $Z = \sum_{\mathbf{y}'} \prod_{c \in \mathcal{C}} \phi_c(\mathbf{y}_c{}')$.

For simplicity of exposition, we focus most of our discussion on *pairwise* Markov networks. We extend our results to higher-order interactions in Sec. 3. A pairwise Markov network is simply a Markov network where all of the cliques involve either a single node or a pair of nodes. Thus, in a pairwise Markov network with edges $E = \{(ij)\}$ $(i < j)$, only nodes and edges are associated with potentials $\phi_i(Y_i)$ and $\phi_{ij}(Y_i, Y_j)$. A pairwise Markov net defines the distribution

$$P_\phi(\mathbf{y}) = \frac{1}{Z} \prod_{i=1}^{N} \phi_i(y_i) \prod_{(ij) \in E} \phi_{ij}(y_i, y_j),$$

where Z is the *partition function* given by $Z = \sum_{\mathbf{y}'} \prod_{i=1}^{N} \phi_i(y_i') \prod_{(ij) \in E} \phi_{ij}(y_i', y_j')$.

The node and edge potentials are functions of the features of the objects $\mathbf{x}_i \in \Re^{d_n}$ and features of the relationships between them $\mathbf{x}_{ij} \in \Re^{d_e}$. In hypertext classification, $\mathbf{x}_i$ might be the counts of the words of the document i, while $\mathbf{x}_{ij}$ might be the words surrounding the hyperlink(s) between documents i and j. The simplest model of dependence of the potentials on the features is a log-linear combination: $\log \phi_i(k) = \mathbf{w}_n^k \cdot \mathbf{x}_i$ and $\log \phi_{ij}(k, l) = \mathbf{w}_e^{k,l} \cdot \mathbf{x}_{ij}$, where $\mathbf{w}_n^k$ and $\mathbf{w}_e^{k,l}$ are label-specific row vectors of node and edge parameters, of size d_n and d_e, respectively. Note that this formulation assumes that all of the nodes in the network share the same set of weights, and similarly all of the edges share the same weights.

We represent an assignment $\mathbf{y}$ as a set of $K \cdot N$ indicators $\{y_i^k\}$, where $y_i^k = I(y_i = k)$. With these definitions, the log of conditional probability $\log P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$

is given by:

$$\sum_{i=1}^{N}\sum_{k=1}^{K}(\mathbf{w}_n^k\cdot\mathbf{x}_i)y_i^k+\sum_{(ij)\in E}\sum_{k,l=1}^{K}(\mathbf{w}_e^{k,l}\cdot\mathbf{x}_{ij})y_i^k y_j^l-\log Z_{\mathbf{w}}(\mathbf{x}).$$

Note that the partition function $Z_{\mathbf{w}}(\mathbf{x})$ above depends on the parameters $\mathbf{w}$ and input features $\mathbf{x}$, but not on the labels y_i's.

For compactness of notation, we define the node and edge weight vectors $\mathbf{w}_n = (\mathbf{w}_n^1, \ldots, \mathbf{w}_n^K)$ and $\mathbf{w}_e = (\mathbf{w}_e^{1,1}, \ldots, \mathbf{w}_e^{K,K})$, and let $\mathbf{w} = (\mathbf{w}_n, \mathbf{w}_e)$ be a vector of all the weights, of size $d = Kd_n + K^2 d_e$. Also, we define the node and edge labels vectors, $\mathbf{y}_n = (\ldots, y_i^1, \ldots, y_i^K, \ldots)^\top$ and $\mathbf{y}_e = (\ldots, y_{ij}^{1,1}, \ldots, y_{ij}^{K,K}, \ldots)^\top$, where $y_{ij}^{k,l} = y_i^k y_j^l$, and the vector of all labels $\mathbf{y} = (\mathbf{y}_n, \mathbf{y}_e)$ of size $L = KN + K^2|E|$. Finally, we define an appropriate $d \times L$ matrix $\mathbf{X}$ such that

$$\log P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}) = \mathbf{w}\mathbf{X}\mathbf{y} - \log Z_{\mathbf{w}}(\mathbf{x}).$$

The matrix $\mathbf{X}$ contains the node feature vectors $\mathbf{x}_i$ and edge feature vectors $\mathbf{x}_{ij}$ repeated multiple times (for each label k or label pair k, l respectively), and padded with zeros appropriately.

A key task in Markov networks is computing the *MAP (maximum a posteriori) assignment* — the assignment $\mathbf{y}$ that maximizes $\log P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x})$. It is straightforward to formulate the MAP inference task as an integer linear program: The variables are the assignments to the nodes y_i^k and edges $y_{ij}^{k,l}$ which must be in the set $\{0, 1\}$, and satisfy linear normalization and agreement constraints. The optimization criterion is simply the linear function $\mathbf{wXy}$, which acorresponds to the log of the unnormalized probability of the assignment $\mathbf{y}$.

In certain cases, we can take this integer program, and approximate it as a linear program by relaxing the integrality constraints on y_i^k, with appropriate constraints. For example, Wainwright *et al.* (2002) provides a natural formulation of this form that is guaranteed to produce integral solutions for triangulated graphs.

3. Associative Markov Networks

We now describe one important subclass of problems for which the above relaxation is particularly useful. These networks, which we call *associative Markov networks (AMNs)*, encode situations where related variables tend to have the same value.

Associative interactions arise naturally in the context of image processing, where nearby pixels are likely to have the same label (Besag, 1986; Boykov et al., 1999b). In this setting, a common approach is to use a *generalized Potts model* (Potts, 1952), which penalizes assignments that do not have the same label across the edge: $\phi_{ij}(k, l) = \lambda_{ij}$, $\forall k \neq l$ and $\phi_{ij}(k, k) = 1$, where $\lambda_{ij} \leq 1$.

For binary-valued Potts models, Greig *et al.* (1989) show that the MAP problem can be formulated as a min-cut in an appropriately constructed graph. Thus, the MAP problem can be solved exactly for this class of models in polynomial time. For $K > 2$, the MAP problem is NP-hard, but a procedure based on a relaxed linear program guarantees a factor 2 approximation of the optimal solution (Boykov et al., 1999b; Kleinberg & Tardos, 1999). Kleinberg and Tardos (1999) extend the multi-class Potts model to have more general edge potentials, under the constraints that negative log potentials $-\log\phi_{ij}(k, l)$ form a metric on the set of labels. They also provide a solution based on a relaxed LP that has certain approximation guarantees.

More recently, Kolmogorov and Zabih (2002) showed how to optimize energy functions containing binary and ternary interactions using graph cuts, as long as the parameters satisfy a certain regularity condition. Our definition of associative potentials below also satisfies the Kolmogorov and Zabih regularity condition for $K = 2$. However, the structure of our potentials is simpler to describe and extend for the multi-class case. We use a linear programming formulation (instead of min-cut) for the MAP inference, which allows us to use the maximum margin estimation framework, as described below. Note however, that we can also use min-cut to perform exact inference on the learned models for $K = 2$ and also in approximate inference for $K > 2$ as in Boykov *et al.* (1999a).

Our associative potentials extend the Potts model in several ways. Importantly, AMNs allow different labels to have different attraction strength: $\phi_{ij}(k, k) = \lambda_{ij}^k$, where $\lambda_{ij}^k \geq 1$, and $\phi_{ij}(k, l) = 1$, $\forall k \neq l$. This additional flexibility is important in many domains, as different labels can have very diverse affinities. For example, foreground pixels tend to have locally coherent values while background is much more varied.

The linear programming relaxation of the MAP problem for these networks can be written as:

$$\max \sum_{i=1}^{N}\sum_{k=1}^{K}(\mathbf{w}_n^k\cdot\mathbf{x}_i)y_i^k+\sum_{(ij)\in E}\sum_{k=1}^{K}(\mathbf{w}_e^{k,k}\cdot\mathbf{x}_{ij})y_{ij}^k \quad (1)$$
$$\text{s.t.}\quad y_i^k \geq 0, \quad \forall i, k; \qquad \sum_k y_i^k = 1, \quad \forall i;$$
$$y_{ij}^k \leq y_i^k, \qquad y_{ij}^k \leq y_j^k, \qquad \forall (ij) \in E, k.$$

Note that we substitute the constraint $y_{ij}^k = y_i^k \wedge y_j^k$ by two linear constraints $y_{ij}^k \leq y_i^k$ and $y_{ij}^k \leq y_j^k$. This works because the coefficient $\mathbf{w}_e^{k,k} \cdot \mathbf{x}_{ij}$ is non-negative and we are maximizing the objective function. Hence,at the optimum $y_{ij}^k = \min(y_i^k, y_j^k)$, which is equivalent to $y_{ij}^k = y_i^k \wedge y_j^k$.

In a second important extension, AMNs admit non-pairwise interactions between variables, with potentials over cliques involving m variables $\phi(y_{i1}, \ldots, y_{im})$. In this case, the clique potentials are constrained to have the same type of structure as the edge potentials: There are K parameters $\phi(k, \ldots, k) = \lambda_{ij}^k$ and the rest of the entries are set to 1. In particular, using this additional expressive power, AMNs allow us to encode the pattern of (soft) transitivity present in many domains. For example, consider the problem of predicting whether two proteins interact (Vazquez et al., 2003); this probability may increase if they *both* interact with another protein. This type of transitivity could be modeled by a ternary clique that has high λ for the assignment with all interactions present.

We can write a linear program for the MAP problem similar to Eq. (1), where we have a variable y_c^k for each clique c and for each label k, which represents the event that all nodes in the clique c have label k:

$$\begin{aligned} \max \quad & \sum_{i=1}^{N}\sum_{k=1}^{K}(\mathbf{w}_n^k \cdot \mathbf{x}_i) y_i^k + \sum_{c \in \mathcal{C}}\sum_{k=1}^{K}(\mathbf{w}_c^k \cdot \mathbf{x}_i) y_c^k \qquad (2) \\ \text{s.t.} \quad & y_i^k \geq 0, \quad \forall i, k; \qquad \sum_k y_i^k = 1, \quad \forall i; \\ & y_c^k \leq y_i^k, \qquad \forall c \in \mathcal{C},\ i \in c, k. \end{aligned}$$

It can be shown that in the binary case, the relaxed linear programs Eq. (1) and Eq. (2) are guaranteed to produce an integer solution when a unique solution exists.

Theorem 3.1 *If $K = 2$, for any objective $\mathbf{wX}$, the linear programs in Eq. (1) and Eq. (2) have an integral optimal solution.*

We omit the proof here due to lack of space. (See the longer version of the paper at `http://cs.stanford.edu/~btaskar/`.) This result states that the MAP problem in binary AMNs is tractable, regardless of network topology or clique size. In the non-binary case ($K > 2$), these LPs can produce fractional solutions and we use a rounding procedure to get an integral solution. In the longer version of the paper, we show that the approximation ratio of the rounding procedure is the inverse of the size of the largest clique (e.g., $\frac{1}{2}$ for pairwise networks). Although artificial examples with fractional solutions can be easily constructed by using symmetry, it seems that in real data such symmetries are often broken. In fact, in all our experiments with $K > 2$ on real data, we never encountered fractional solutions.

4. Max Margin Estimation

We now consider the problem of training the weights $\mathbf{w}$ of a Markov network given a labeled training instance $(\mathbf{x}, \hat{\mathbf{y}})$. For simplicity of exposition, we assume that we have only a single training instance; the extension to the case of multiple instances is entirely straightforward. Note that, in our setting, a single training instance actually contains multiple objects. For example, in the hypertext domain, an instance might be an entire website, containing many inter-linked webpages.

The M^{3}N Framework. The standard approach of learning the weights $\mathbf{w}$ given $(\mathbf{x}, \hat{\mathbf{y}})$ is to maximize the $\log P_{\mathbf{w}}(\hat{\mathbf{y}} \mid \mathbf{x})$, with an additional regularization term, which is usually taken to be the squared-norm of the weights $\mathbf{w}$ (Lafferty et al., 2001). An alternative method, recently proposed by Taskar *et al.* (2003a), is to maximize the margin of confidence in the true label assignment $\hat{\mathbf{y}}$ over any other assignment $\mathbf{y} \neq \hat{\mathbf{y}}$. They show that the margin-maximization criterion provides significant improvements in accuracy over a range of problems. It also allows high-dimensional feature spaces to be utilized by using the kernel trick, as in support vector machines. The maximum margin Markov network (M^{3}N) framework forms the basis for our work, so we begin by reviewing this approach.

As in support vector machines, the goal in an M^{3}N is to maximize our confidence in the true labels $\hat{\mathbf{y}}$ relative to any other possible joint labelling $\mathbf{y}$. Specifically, we define the gain of the true labels $\hat{\mathbf{y}}$ over another possible joint labelling $\mathbf{y}$ as:

$$\log P_{\mathbf{w}}(\hat{\mathbf{y}} \mid \mathbf{x}) - \log P_{\mathbf{w}}(\mathbf{y} \mid \mathbf{x}) = \mathbf{wX}(\hat{\mathbf{y}} - \mathbf{y}).$$

In M^{3}Ns, the desired gain takes into account the number of labels in $\mathbf{y}$ that are misclassified, $\Delta(\hat{\mathbf{y}}, \mathbf{y})$, by scaling linearly with it:

$$\max \quad \gamma \quad \text{s.t.} \quad \mathbf{wX}(\hat{\mathbf{y}} - \mathbf{y}) \geq \gamma \Delta(\hat{\mathbf{y}}, \mathbf{y}); \;\; ||\mathbf{w}||^2 \leq 1.$$

Note that the number of incorrect node labels $\Delta(\hat{\mathbf{y}}, \mathbf{y})$ can also be written as $N - \hat{\mathbf{y}}_n^\top \mathbf{y}_n$. (Whenever $\hat{y}_i$ and y_i agree on some label k, we have that $\hat{y}_i^k = 1$ and $y_i^k = 1$, adding 1 to $\hat{\mathbf{y}}_n^\top \mathbf{y}_n$.) By dividing through by γ and adding a slack variable for non-separable data, we obtain a quadratic program (QP) with exponentially many constraints:

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathbf{w}||^2 + C\xi \qquad (3) \\ \text{s.t.} \quad & \mathbf{wX}(\hat{\mathbf{y}} - \mathbf{y}) \geq N - \hat{\mathbf{y}}_n^\top \mathbf{y}_n - \xi, \;\; \forall \mathbf{y} \in \mathcal{Y}. \end{aligned}$$

This QP has a constraint for every possible joint assignment $\mathbf{y}$ to the Markov network variables, resulting in an exponentially-sized QP. Taskar *et al.* show how

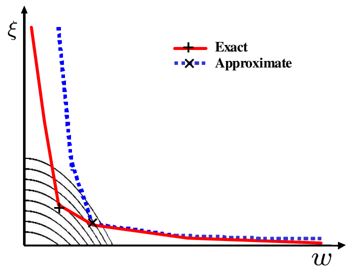


Figure 1. Exact and approximate constraints on the max-margin quadratic program. The solid red line represents the constraints imposed by integer $\mathbf{y}$'s, whereas the dashed blue line represents the stronger constraints imposed by the larger set of fractional $\mathbf{y}$'s. The fractional constraints may coincide with the integer constraints in some cases, and be more stringent in others. The parabolic contours represent the value of the objective function.

structure in the dual of this QP can be exploited to allow an efficient solution when the underlying network has low treewidth.

M^{3}N relaxations.

As an alternative to the approach of Taskar *et al.*, we now derive a more generally applicable approach for exploiting structure and relaxations in max-margin problems. As our first step, we replace the exponential set of linear constraints in the max-margin QP of Eq. (3) with the single equivalent non-linear constraint:

$$\mathbf{wX\hat{y}} - N + \xi \geq \max_{\mathbf{y} \in \mathcal{Y}} \quad \mathbf{wXy} - \hat{\mathbf{y}}_n^\top \mathbf{y}_n.$$

This non-linear constraint essentially requires that we find the assignment $\mathbf{y}$ to the network variables which has the highest probability relative to the parameterization $\mathbf{wX} - \hat{\mathbf{y}}_n^\top$. Thus, optimizing the max-margin QP contains the MAP inference task as a component.

As we discussed earlier, we can formulate the MAP problem as an integer program, and then relax it into a linear program. Inserting the relaxed LP into the QP of Eq. (3), we obtain:

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathbf{w}||^2 + C\xi & (4) \\ \text{s.t.} \quad & \mathbf{wX\hat{y}} - N + \xi \geq \max_{\mathbf{y} \in \mathcal{Y}'} \quad \mathbf{wXy} - \hat{\mathbf{y}}_n^\top \mathbf{y}_n. \end{aligned}$$

where $\mathcal{Y}'$ is the space of all legal fractional values for $\mathbf{y}$. In effect, we obtain a QP with a continuum of constraints, one for every fractional assignment to $\mathbf{y}$.

It follows that, in cases where the relaxed LP is guaranteed to provide integer solutions, the integer and relaxed constraint sets coincide, so that the approximate QP is computing precisely the optimal max-margin solution. In the general case, the linear relaxation strengthens the constraints on $\mathbf{w}$ by potentially adding constraints corresponding to fractional assignments $\mathbf{y}$. Fig. 1 shows how the relaxation of the max subproblem reduces the feasible space of $\mathbf{w}$ and ξ. Note that for every setting of the weights $\mathbf{w}$ that produces fractional solutions for the LP relaxation, the approximate constraints are tightened because of the additional fractional assignments $\mathbf{y}$. In this case, the fractional MAP solution is better than any integer solution, including $\hat{\mathbf{y}}$, thereby driving up the corresponding slack ξ. By contrast, for weights $\mathbf{w}$ for which the MAP LP is integer-valued, the margin has the standard interpretation as the difference between the probability of $\hat{\mathbf{y}}$ and the MAP $\mathbf{y}$ (according to $\mathbf{w}$). As the objective includes a penalty for the slack variable, intuitively, minimizing the objective tends to drive the weights $\mathbf{w}$ away from the regions where the solutions to the MAP LP are fractional.

While this insight allows us to replace the MAP integer program within the QP with a linear program, the resulting QP does not appear tractable. However, here we can exploit fundamental properties of linear programming duality (Bertsimas & Tsitsiklis, 1997). Assume that our relaxed LP for the inference task has the form:

$$\max_{\mathbf{y}} \quad \mathbf{wBy} \quad \text{s.t.} \quad \mathbf{y} \geq 0, \quad \mathbf{Ay} \leq \mathbf{b}. \qquad (5)$$

for some polynomial-size $\mathbf{A}, \mathbf{B}, \mathbf{b}$. (For example, Eq. (1) and Eq. (2) can be easily written in this compact form.) The dual of this LP is given by:

$$\min_{\mathbf{z}} \quad \mathbf{b}^\top \mathbf{z} \quad \text{s.t.} \quad \mathbf{z} \geq 0, \mathbf{A}^\top \mathbf{z} \geq (\mathbf{wB})^\top. \qquad (6)$$

When the relaxed LP is feasible and bounded, the value of Eq. (6) provides an upper bound on the primal that achieves the same value as the primal at its minimum. If we substitute Eq. (6) for Eq. (5) in the QP of Eq. (4), we obtain a quadratic program over $\mathbf{w}$, ξ and $\mathbf{z}$ with polynomially many linear constraints:

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathbf{w}||^2 + C\xi & (7) \\ \text{s.t.} \quad & \mathbf{wX\hat{y}} - N + \xi \geq \mathbf{b}^\top \mathbf{z}; \\ & \mathbf{z} \geq 0, \quad \mathbf{A}^\top \mathbf{z} \geq (\mathbf{wB})^\top. \end{aligned}$$

Our ability to perform this transformation is a direct consequence of the connection between the max-margin criterion and the MAP inference problem. The transformation is useful whenever we can solve or approximate MAP using a compact linear program.

5. Max Margin AMNs

The transformation described in the previous section applies to any situation where the MAP problem can be effectively approximated as a linear program. In particular, the LP relaxation of Eq. (1) provides us with precisely the necessary building block to provide an effective solution for the QP in Eq. (4) for the case of AMNs. As we discussed, the MAP problem is precisely the max subproblem in this QP. In the case of AMNs, this max subproblem can be replaced with the relaxed LP of Eq. (1). In effect, we are replacing the exponential constraint set — one which includes a constraint for every discrete $\mathbf{y}$, with an infinite constraint set — one which includes a constraint for every continuous vector $\mathbf{y}$ in

$$\mathcal{Y}' = \{\mathbf{y} : y_i^k \geq 0;\ \sum_k y_i^k = 1;\ y_{ij}^k \leq y_i^k;\ y_{ij}^k \leq y_j^k\}$$

as defined in Eq. (1).

Stating the AMN restrictions in terms of the parameters $\mathbf{w}$, we require that $\mathbf{w}_e^{k,l} = 0,\ \forall k \neq l$ and $\mathbf{w}_e^{k,k} \cdot \mathbf{x}_{ij} \geq 0$. To ensure that $\mathbf{w}_e^{k,k} \cdot \mathbf{x}_{ij} \geq 0$, we simply assume (without loss of generality) that $\mathbf{x}_{ij} \geq 0$, and constrain $\mathbf{w}_e^{k,k} \geq 0$. Incorporating this constraint, we obtain our basic AMN QP:

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathbf{w}||^2 + C\xi && (8) \\ \text{s.t.} \quad & \mathbf{w}\mathbf{X}\hat{\mathbf{y}} - N + \xi \geq \max_{\mathbf{y}\in\mathcal{Y}'} \mathbf{w}\mathbf{X}\mathbf{y} - \hat{\mathbf{y}}_n \cdot \mathbf{y}_n; \\ & \mathbf{w}_e \geq 0. \end{aligned}$$

We can now transform this QP as specified in Eq. (7), by taking the dual of the LP used to represent the interior max. Specifically, $\max_{\mathbf{y}\in\mathcal{Y}'} \mathbf{w}\mathbf{X}\mathbf{y} - \hat{\mathbf{y}}_n \cdot \mathbf{y}_n$ is a feasible and bounded linear program in $\mathbf{y}$, with a dual given by:

$$\begin{aligned} \min \quad & \sum_{i=1}^{N} z_i && (9) \\ \text{s.t.} \quad & z_i - \sum_{(ij),(ji)\in E} z_{ij}^k \geq \mathbf{w}_n^k \cdot \mathbf{x}_i - \hat{y}_i^k, \quad \forall i, k; \\ & z_{ij}^k + z_{ji}^k \geq \mathbf{w}_e^{k,k} \cdot \mathbf{x}_{ij}, \quad z_{ij}^k, z_{ji}^k \geq 0, \quad \forall (ij) \in E, k. \end{aligned}$$

In the dual, we have a variable z_i for each normalization constraint in Eq. (1) and variables z_{ij}^k, z_{ji}^k for each of the inequality constraints.

Substituting this dual into Eq. (8), we obtain:

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathbf{w}||^2 + C\xi && (10) \\ \text{s.t.} \quad & \mathbf{w}\mathbf{X}\hat{\mathbf{y}} - N + \xi \geq \sum_{i=1}^{N} z_i; \quad \mathbf{w}_e \geq 0; \\ & z_i - \sum_{(ij),(ji)\in E} z_{ij}^k \geq \mathbf{w}_n^k \cdot \mathbf{x}_i - \hat{y}_i^k, \quad \forall i, k; \\ & z_{ij}^k + z_{ji}^k \geq \mathbf{w}_e^{k,k} \cdot \mathbf{x}_{ij}, \quad z_{ij}^k, z_{ji}^k \geq 0, \quad \forall (ij) \in E, k. \end{aligned}$$

For $K = 2$, the LP relaxation is exact, so that Eq. (10) learns *exact* max-margin weights for Markov networks of *arbitrary* topology. For $K > 2$, the linear relaxation leads to a strengthening of the constraints on $\mathbf{w}$ by potentially adding constraints corresponding to fractional assignments $\mathbf{y}$. Thus, the optimal choice $\mathbf{w}, \xi$ for the original QP may no longer be feasible, leading to a different choice of weights. However, as our experiments show, these weights tend to do well in practice.

The dual of Eq. (10) provides some insight into the structure of the problem:

$$\begin{aligned} \max \quad & \sum_{i=1}^{N}\sum_{k=1}^{K}(1 - \hat{y}_i^k)\mu_i^k && (11) \\ & -\frac{1}{2}\sum_{k=1}^{K}\left|\left|\sum_{i=1}^{N}\mathbf{x}_i(C\hat{y}_i^k - \mu_i^k)\right|\right|^2 \\ & -\frac{1}{2}\sum_{k=1}^{K}\left|\left|\lambda_e^k + \sum_{(ij)\in E}\mathbf{x}_{ij}(C\hat{y}_{ij}^k - \mu_{ij}^k)\right|\right|^2 \\ \text{s.t.} \quad & \mu_i^k \geq 0, \quad \forall i, k; \qquad \sum_k \mu_i^k = C, \quad \forall i; \\ & \mu_{ij}^k \geq 0, \quad \mu_{ij}^k \leq \mu_i^k, \quad \mu_{ij}^k \leq \mu_j^k, \quad \forall (ij) \in E, k; \\ & \lambda_e \geq 0. \end{aligned}$$

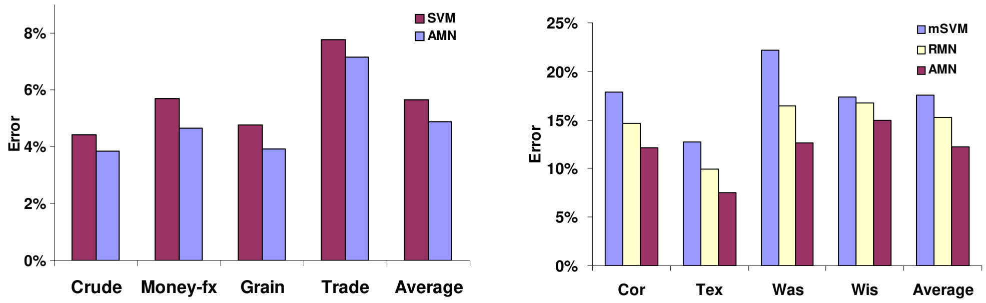
As in the original $\mathrm{M^3N}$ optimization, the dual variables have an intuitive probabilistic interpretation. In the binary case, the set of the variables μ_i^k, μ_{ij}^k corresponds to marginals of a distribution (normalized to C) over the possible assignments $\mathbf{y}$. (This assertion follows from taking the dual of the original exponential size QP in Eq. (3).) Then the constraints that $\mu_{ij}^k \leq \mu_i^k$ and $\mu_{ij}^k \leq \mu_j^k$ can be explained by the fact that $P(y_i = y_j = k) \leq P(y_i = k)$ and $P(y_i = y_j = k) \leq P(y_j = k)$ for any distribution $P(\mathbf{y})$. For $K > 2$, the set of the variables μ_i^k, μ_{ij}^k may not correspond to a valid distribution.

The primal and dual solution are related by:

$$\mathbf{w}_n^k = \sum_{i=1}^{N}\mathbf{x}_i(C\hat{y}_i^k - \mu_i^k), \qquad (12)$$

$$\mathbf{w}_e^{k,k} = \lambda_e^k + \sum_{(ij)\in E}\mathbf{x}_{ij}(C\hat{y}_{ij}^k - \mu_{ij}^k). \qquad (13)$$

One important consequence of these relationships is that the node parameters are all support vector expansions. Thus, the terms in the constraints of the form $\mathbf{w}_n\mathbf{x}$ can all be expanded in terms of dot products $\mathbf{x}_i^\top\mathbf{x}_j$; the objective ($||\mathbf{w}||^2$) can be expanded similarly. Therefore, we can use kernels $K(\mathbf{x}_i, \mathbf{x}_j)$ to define node parameters. Unfortunately, the positivity constraint on the edge potentials, and the resulting λ_e^k dual variable in the expansion of the edge weight, prevent the edge parameters from being kernelized in a similar way.



(a) (b)

Figure 2. (a) Comparison of test error of SVMs and AMNs on four categories of Reuters articles, averaged over 7-folds; (b) Comparison of test error of SVMs, RMNs and AMNs on four WebKB sites.

6. Experimental Results

We evaluated our approach on two text classification domains, of very different structure.

Reuters. We ran our method on the ModApte set of the Reuters-21578 corpus. We selected four categories containing a substantial number of documents: crude, grain, trade, and money-fx. We eliminated documents labeled with more than one category, and represented each document as a bag of words. The resulting dataset contained around 2200 news articles, which were split into seven folds where the articles in each fold occur in the same time period. The reported results were obtained using seven-fold cross-validation with a training set size of $\sim$ 200 documents and a test set size of $\sim$ 2000 documents.

The baseline model is a linear kernel SVM using a bag of words as features. Since we train and test on articles in different time periods, there is an inherent distribution drift between our training and test sets, which hurts the SVM's performance. For example, there may be words which, in the test set, are highly indicative of a certain label, but are not present in the training set at all since they were very specific to a particular time period (see (Taskar et al., 2003b)).

Our AMN model uses the text similarity of two articles as an indicator of how likely they are to have the same label. The intuition is that two documents that have similar text are likely to share the same label in any time period, so that adding associative edges between them would result in better classification. Such positive correlations are exactly what AMNs represent. In our model, we linked each document to its two closest documents as measured by TF-IDF weighted cosine distance. The TF-IDF score of a term was computed as: $(1 + \log tf) \log \frac{N}{df}$ where tf is the term frequency, N is the number of total documents, and df is the document frequency. The node features were simply the words in the article corresponding to the node. Edge features included the actual TF-IDF weighted cosine distance, as well as the bag of words consisting of union of the words in the linked documents.

We trained both models (SVM and AMN) to predict one category vs. all remaining categories. Fig. 2(a) shows that the AMN model achieves a 13.5% average error reduction over the baseline SVM, with improvement in every category. Applying a paired t-test comparing the AMN and SVM over the 7 folds in each category, crude, trade, grain, money-fx, we obtained p-values of 0.004897, 0.017026, 0.012836, 0.000291 respectively. These results indicate that the positive interactions learned by the AMN allow us to correct for some of the distribution drift between the training and test sets.

Hypertext. We tested AMNs on collective hypertext classification, using the variant of the WebKB dataset (Craven et al., 1998) used by Taskar *et al.* (2002). This data set contains web pages from four different Computer Science departments: Cornell, Texas, Washington, and Wisconsin. Each page is labeled as one of course, faculty, student, project, other. Our goal in this task is to exploit the additional structured information in hypertext using AMNs.

Our flat model is a multiclass linear-kernel SVM predicting categories based on the text content of the webpage. The words are represented as a bag of words. For the AMN model, we used the fact that a webpage's internal structure can be broken up into disjoint *sections*. For example, a faculty webpage might have one section that discusses research, with a list of links to relevant research projects, another section with links to student webpages, etc. Intuitively, if we have links to two pages in the same section, they are likely have the same topic. As AMNs capture precisely this type of positive correlation, we added edges between pages that appear as hyperlinks in the same section of another page. The node features for the AMN model are the same as for the multiclass SVM.

In performing the experiments we train on the pages from three of the schools in the dataset and test on the remaining one. The results, shown in Fig. 2(b), demonstrate a 30% relative reduction in test error as a result of modeling the positive correlation between pages in the AMN model. The improvement is present when testing on each of the schools. We also trained the same AMN model using the RMN approach of Taskar *et al.* (2002). In this approach, the Markov network is trained to maximize the conditional log-likelihood, using loopy belief propagation (Yedidia et al., 2000) for computing the posterior probabilities needed for optimization. Due to the high connectivity in the network, the algorithm is not exact, and not guaranteed to converge to the true values for the posterior distribution. In our results, RMNs achieve a worse test error than AMNs. We note that the learned AMN weights never produced fractional solutions when used for inference, which suggests that the optimization successfully avoided problematic parameterizations of the network, even in the case of the non-optimal multi-class relaxation.

7. Conclusion

In this paper, we provide an algorithm for max-margin training of associative Markov networks, a subclass of Markov networks that allows only positive interactions between related variables. Our approach relies on a linear programming relaxation of the MAP problem, which is the key component in the quadratic program associated with the max-margin formulation. We thus provide a polynomial time algorithm which approximately solves the maximum margin estimation problem for any associative Markov network. Importantly, our method is guaranteed to find the optimal (margin-maximizing) solution for all binary-valued AMNs, regardless of the clique size or the connectivity. To our knowledge, this algorithm is the first to provide an effective learning procedure for Markov networks of such general structure.

Our results in the binary case rely on the fact that the LP relaxation of the MAP problem provides exact solutions. In the non-binary case, we are not guaranteed exact solutions, but we can prove constant-factor approximation bounds on the MAP solution returned by the relaxed LP. It would be interesting to see whether these bounds provide us with guarantees on the quality (e.g., the margin) of our learned model.

The class of associative Markov networks appears to cover a large number of interesting applications. We have explored only two such applications in our experimental results, both in the text domain. It would be very interesting to consider other applications, such as image segmentation, extracting protein complexes from protein-protein interaction data, or predicting links in relational data.

However, despite the prevalence of fully associative Markov networks, it is clear that many applications call for repulsive potentials. For example, the best classification accuracy on the WebKB hypertext data set is obtained in a maximum margin framework (Taskar et al., 2003a), when we allow repulsive potentials on linked webpages (representing, for example, that students tend not to link to pages of students). While clearly we cannot introduce fully general potentials into AMNs without running against the NP-hardness of the general problem, it would be interesting to see whether we can extend the class of networks we can learn effectively.

References

Bach, F., & Jordan, M. (2001). Thin junction trees. *NIPS*.

Bertsimas, D., & Tsitsiklis, J. (1997). *Introduction to linear programming*. Athena Scientific.

Besag, J. E. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society B*, *48*.

Bouman, C., & Shapiro, M. (1994). A multiscale random field model for bayesian image segmentation. *IP*, *3*.

Boykov, Y., Veksler, O., & Zabih, R. (1999a). Fast approximate energy minimization via graph cuts. *ICCV*.

Boykov, Y., Veksler, O., & Zabih, R. (1999b). Markov random fields with efficient approximations. *CVPR*.

Chakrabarti, S., Dom, B., & Indyk, P. (1998). Enhanced hypertext categorization using hyperlinks. *SIGMOD*.

Craven, M., DiPasquo, D., Freitag, D., McCallum, A., Mitchell, T., Nigam, K., & Slattery, S. (1998). Learning to extract symbolic knowledge from the world wide web. *Proc AAAI98* (pp. 509–516).

Greig, D. M., Porteous, B. T., & Seheult, A. H. (1989). Exact maximum a posteriori estimation for binary images. *J. R. Statist. Soc. B*, *51*.

Kleinberg, J., & Tardos, E. (1999). Approximation algorithms for classification problems with pairwise relationships: Metric labeling and markov random fields. *FOCS*.

Kolmogorov, V., & Zabih, R. (2002). What energy functions can be minimized using graph cuts? *PAMI*.

Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*.

Potts, R. B. (1952). Some generalized order-disorder transformations. *Proc. Cambridge Phil. Soc.*, *48*.

Taskar, B., Abbeel, P., & Koller, D. (2002). Discriminative probabilistic models for relational data. *UAI*.

Taskar, B., Guestrin, C., & Koller, D. (2003a). Max margin markov networks. *Proc. NIPS*.

Taskar, B., Wong, M., & Koller, D. (2003b). Learning on the test data: Leveraging unseen features. *Proc. ICML*.

Vazquez, A., Flammini, A., Maritan, A., & Vespignani, A. (2003). Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, *6*.

Wainwright, M., Jaakkola, T., & Willsky, A. (2002). Map estimation via agreement on (hyper)trees: Message-passing and linear programming approaches. *Allerton Conference on Communication, Control and Computing*.

Yedidia, J., Freeman, W., & Weiss, Y. (2000). Generalized belief propagation. *NIPS*.